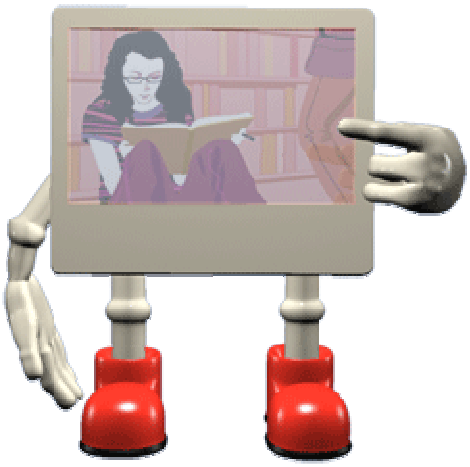
Instructional Design Approaches

Instructional Design



Two important considerations of web based training environments were determined by Horton (2001) clearly. One is ideal learner and the second one is the ideal course. The relationship between two components is that web based courses generally require ideal learner in order to work effectively. In addition to ideal learner, an ideal course should fulfill needs of ideal students. The question is now what are the main characteristics of the ideal learner and the ideal course. ?

The ideal learner

Technology does not have same impact on people because of different individual characteristics. Certain groups have a chance to get more benefits form technological resources. It is, therefore, important to determine who can be more successful in web based training environments. Horton (2001) expresses that the ideal learner who:

- Opens to learn independently and view learning prohibitively,
- Is self-disciplined, have good time management, enjoy working alone,
- Is good at writing skills,
- Has enough competencies in computing skills,
- Has positive attitude toward technology in business and learning,
- Needs to acquire a new knowledge but it can be realized easily,
- Is so relaxed while there is any technical problem and can cope with these problems,
- Has definite goal,
- Is moderately capable of field and know basic concepts and facts.

Ideal Learner is necessary but not sufficient condition in WBT. A well structured course could support these persons with desired consequences. It is possible to acquire when there is an ideal course.

The ideal course

In an ideal course, teaching is efficient and precise objectives are well defined. Therefore, learner acquires new knowledge by spending less time. Objectives are also singing for learner who can predict benefits of course. Designers can use as much as possible models in web based training courses. Important point is satisfaction and learning outcomes of participants. The best way to shape an ideal course is determining needs of learners.

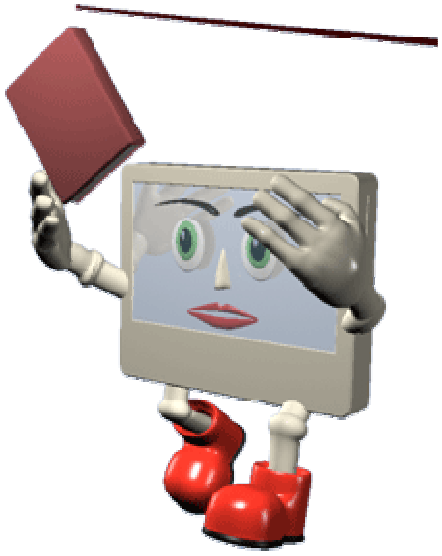
Once instructional designers can determine learner and course, development process of WBT is another important concern. In development process, software development models are suitable guidance. In some situations, one development model can be enough to complete project, whereas in some situations, it is impossible to finish project with one model. Therefore, combination of software development models also can be implemented. In this section, major software development models are going to be presented.

The Three Priorities for Training Success

In e-learning environments, there too many things that can easily influence focus in negative way. It is because of the nature of e-learning that it can provide endless array of opportunities. Therefore, one can find the both most and least useful knowledge at the same time. Hence, designers of e-learning courses are responsible of successful e-learning environments. The main three things are important to make successful designs (Allen, 2003).

a. Designers should ensure that learners are highly motivated to learn. Motivation is essential to learning. Moreover, in e-learning, it is not only essential but also required. Motivation can support persistence and participation to activities. Motivation generally leads to person find a way to learn new knowledge. Although motivation is a intrinsic action of a person, your training program must provide some motivational things to increase personals motivation extrinsically.
b. Designers should guide learners to appropriate content. High motivational persons always seek for some new information. Environment cannot provide reciprocal activities to motivation. Yet, it hinders motivation rather than supporting. E-learning course should provide understanding content to participants. It could be sometimes structured content for low ability students, whereas it could be more complex and required to more challenge for high ability students. In fact, it is based on combination of students' characteristics and their needs. Development of e-learning content is the most vital concern. Designer should be capable of shaping appropriate content to their target group. In a summary, your students should be really satisfied from the content of your e-learning course in the dimension of both cognitively, affectively, and any other kinds of learning outcomes.
c. Designers should provide meaningful and memorable learning experiences. Motivation and good prepared materials are essential but they are not sufficient properties of e-learning course. Learning experiences could be sustainable to all situations if they are required. In other words, they could be used in every needed situation. It is about transfer and utilization of training. To support transfer of training at the maximum level, training activities are easily memorable and meaningful. If a learner does not understand anything from your training, it is non-sense thing to carry on. Designers should consider the least skilled groups while they are instructing new information. Therefore, to teach new information becomes possible. Another point is providing multi channels to teach something. Therefore, designer can present new knowledge to every possible kind of students' types. Meaningfulness of e-learning environment should be persistence. Meaningful experiences are required but they cannot support completely successful e-learning courses. Each meaningful experience should be memorable so they can be used in job settings where all training programs are designed for that purpose. Allen (2003) provided some suggestions to make e-learning experiences memorable. Designers can use;
   a. Interesting contexts and novel situations,
   b. Real-world or authentic environments,
   c. Problem-solving scenarios,
   d. Simulations,
   e. Risk and consequences,
   f. Engaging themes,
   g. Engaging media and interface elements,
   h. Drill and practice,
   i. Humor

Ensuring That Learners Are Highly Motivated to Learn



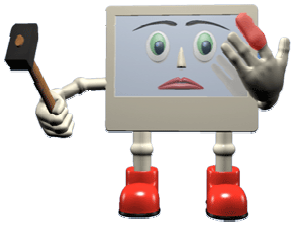
Motivation is essential to learning because it energizes learner attention, persistence, and participation in learning activities. Highly motivated learners will find a way to learn. They will even be creative, if necessary, to find sources of information, best practices, and so on. They will support each other, exchanging information and teaming up to find any missing pieces. If your training program gets you only this far, you've probably already won the toughest battle. Now you can go on to make your performance solution more cost-effective by addressing the remaining two priorities.

Guiding Learners to Appropriate Content

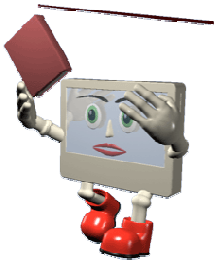
Highly motivated learners are eager to get their hands on anything that will help them learn. In response to this motivation, it is important to provide appropriate material in a timely manner, before the motivation wanes.

There's much more to this than meets the eye, and it's not just the challenge of creating clear, understandable content—which does take expertise, make no mistake about that. Making sure content is appropriate for an individual learner means either providing excellent indexing (navigation) to help learners identify appropriate material, or applying an assessment that will determine what each learner needs and is ready to use. In either case, e-learning is often a cost-effective means of providing content access.

Providing Meaningful and Memorable Learning Experiences



Motivation and good materials are sometimes not enough to enable people to perform at the necessary level or to ready them fast enough. In many cases, there is a wide chasm between building motivation and providing good resource materials and achieving sustained performance competencies. Preparing effective learning experiences is then essential.



One can read extensively about delivering good speeches, handling a difficult customer, or managing a complex project, for example, but guidance and practice will still be necessary to reach needed levels of proficiency. In many cases, it is preferable for learners to make mistakes in a learning environment, where guidance is available and errors are harmless, rather than on the job, where thorough guidance may be more difficult to provide and errors could be damaging to people, equipment, materials, or business.

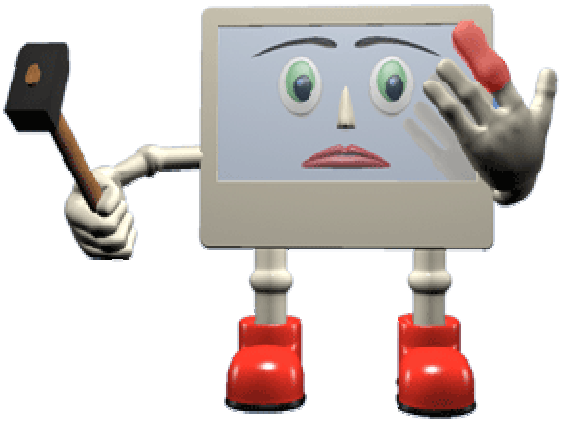
Meaningful Experience

If a learner doesn't understand, then that learner will not gain from the experience. This is instructional failure. Designers of single-channel deliveries for multiple learners, such as classroom presentations, must decide whether to speak to the least able learners (in hopes that other learners will tune in at the appropriate points) or target the average learner (in hopes that unprepared learners will catch up and others will wait patiently for something of value). The approach often results in many learning casualties.

Further, if learners don't see the meaningful implications of learning prescribed tasks, such as the tasks' applicability to the work they do or the advantages of new processes over the ones they currently use, the learning experience is also likely to be of little avail. When different learners perform different functions, it is important to help learners see the relevance of the training to their respective responsibilities. You shouldn't just assume that they'll see it.

Well-designed e-learning has the means to be continuously meaningful for each learner. It can be sensitive to learner performance, identify levels of need and readiness, select appropriate activities, and engage learners in experiences that are likely to be meaningful.

Memorable Experience



If meaningful experiences and the knowledge they convey are easily forgotten (as in a day or two after a posttest), or if learners don't think to apply them in appropriate on-the-job situations, they might as well not have occurred. Time spent in training is expensive to employers. It is not usually the goal of training to simply give workers some enjoyable time off, only to return to the job with no improved abilities. Thankfully, e-learning has many ways to make experiences memorable, such as using:

- Interesting contexts and novel situations
- Real-world or authentic environmentsProblem-solving scenarios
- Simulations

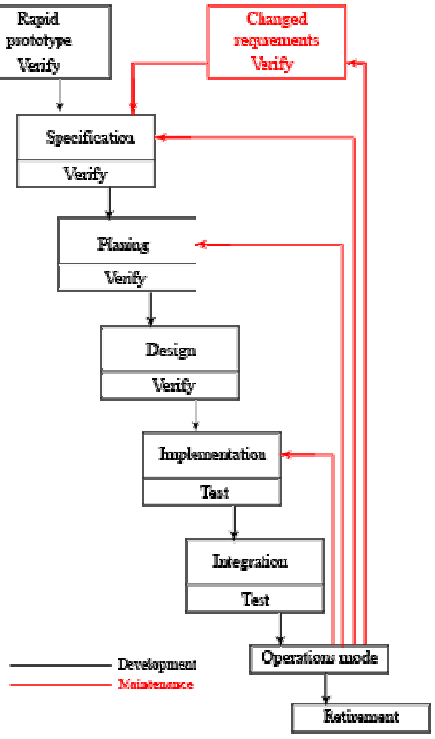
- Risk and consequences
- Engaging themes
- Engaging media and interface elements
- Drill and practice
- Humor

Software Design Methods

Different application development tasks often produce varied design problems and hence require the use of combinations of approaches and models. Among the models briefly covered in this chapter are: Prototyping Model, Rapid Application Development (RAD) Model, Dynamic System Development Method (DSDM), Boehm's Spiral Model, Incremental Model, Component Assembly Model, Concurrent Development Model and Unified Modeling Language (UML).

Protyping Model

In this model, software is developed as partial products. Later, these partial products are used to evaluate and test some aspects of systems requirements. It is useful when customers can participate in design process. Moreover, customers can enter some inputs to product. Hence, some misunderstandings and ambiguities can be solved in effective and efficient ways.
This model starts with gathering information and quick test, and then a prototype are produced and given to customer for evaluation. Once customer is happy with product, either evolutionary process can be implemented to extend the prototype to fulfill needs of whole systems or prototype can be discarded completely.



Applied from Dastbaz, M. (2002). Chapter 8: Overview of Software Design Methods. Designing Interactive Multimedia Systems. McGraw-Hill: London.

Rapid Application Development (RAD) Model

There are no distinctive differences between rapid prototyping and rapid application development. In prototyping model, the aim of development is partially working systems. However, RAD aims to develop fully working system. Their common point is building a working system rapidly. RAD requires more man power than prototyping model.

Dynamic System Development Method (DSDM)

According to Stapleton (1998, as cited in Dastbaz, 2002) the DSDM is based on nine underlying principles including:

- Active user involvement in the design and development process;
- DSDM teams are empowered to make decisions;
- The focus of the DSDM is on frequent product delivery;

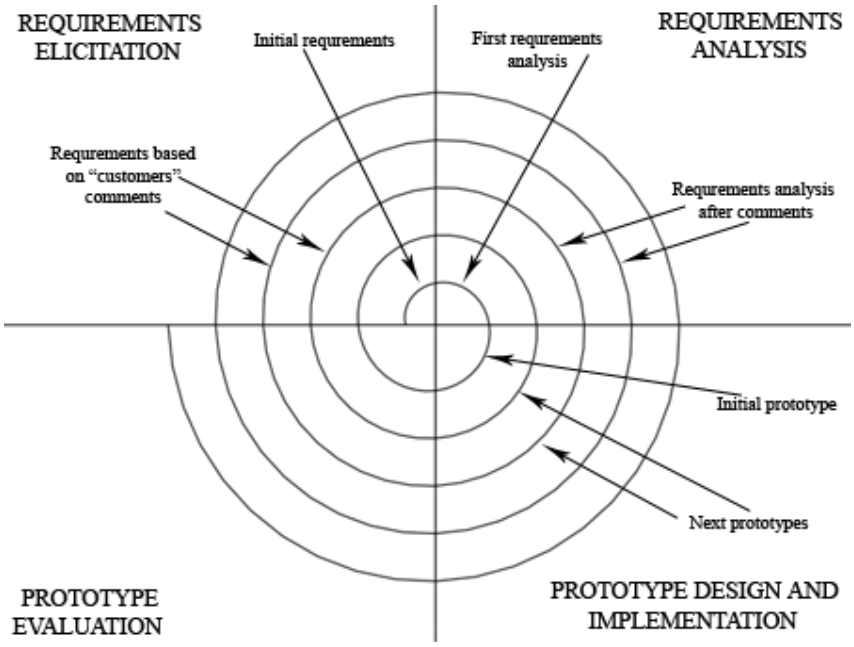
- The essential criterion for accepting a task is its fitness for business purpose;
- DSDM works on the principle of iterative and incremental developments which allows user's feedback;
- Within a DSDM approach all changes during development are reversible;
- Requirement specifications are agreed at the initial stage at high level of decision making;
- Testing is an integral part of the development life cycle;
- DSDM envisages a collaborative and co-operative approach amongst all interested parties.

Main advantages of this method is that time can be saved considerable amount in whole development of software and like prototyping and rapid application development, software products are delivered rapidly.

Boehm's Spiral Model

In this model, software is developed based on incremental iterations. At first iteration, software is developed on paper and in each increasing iteration it is closer to full version.

In contrast to other models, spiral model continue through the life of software. The spiral model uses both systematic step – by – step approach in classical life cycle model and iterative framework. There are no limits to number of iterations. In every iteration step, program can be better because of some newly discovered errors.



Applied from Dastbaz, M. (2002). Chapter 8: Overview of Software Design Methods. Designing Interactive Multimedia Systems. McGraw-Hill: London.

Incremental Model

Incremental Model is a combination or iterative and prototyping approach. Software product is developed in each increment. Initial increments includes basic feature of the system. As the time progresses, more sophisticated version of software are released. Incremental Model conducts some initial analysis to identify some related tasks that are required to be completed for the process. Incremental model provides an opportunity to designers to get feedback about software while it is being developed

Component Assembly Model

This model is similar to spiral and incremental model. However, this model based on object-oriented approach. Components and classes are developed in any stage of the process and then they are stored in a library for re-use when it is needed. It is possible to use some classes in the first phase of iteration. Designer can develop additional classes based on classes used in first iteration. Process follows spiral model and ultimately re-enters the component assembly iteration. The major crucial advantages of this model are reusable components or classes that cause reduction of costs.

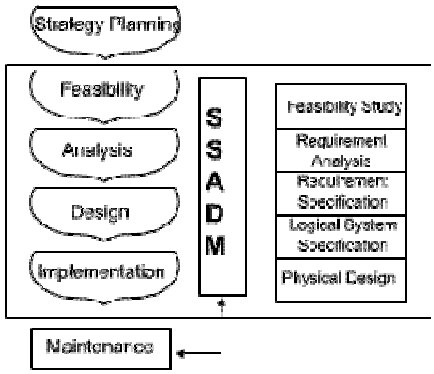
Structured System Analysis & Design Methodology (SSADM)

SSDM has six phases that includes own sub-phases. These are;

- Analysis Phase: It aims at identifying current and ideal data

flow and a logical structure.

- Specification Requirement Phase: At this phase, determining audit control and security and extending logical data structure. Required data flow and the process outline created.
- User Selection Phase: Users' options are created. Performance objectives are selected and set.



Applied from Dastbaz, M. (2002). Chapter 8: Overview of Software Design Methods. Designing Interactive Multimedia Systems. McGraw-Hill: London.

- Detailed Data Design Phase: At this stage, there are three sub-stages. These are creation of detailed logical data structure, building of composite data structure, and setting up of data dictionary.
- Detailed Procedure Design Phase: Physical design control and the design manual procedures are implemented.
- Physical Design Control Phase: System test plan is designed and program specifications are created.

Formal Methods

The proponents of formal method state that mathematics provides accurate results, which no other tools can provide. Formal methods do not make extensive use of text and diagrams as in non-formal methods therefore they are more precise and less prone to ambiguity.

Opponents perceive formal methods to be rather difficult than graphical and textual methods, and, therefore believe that it should be used mainly in systems where accuracy is of paramount importance, such as, systems, which may have an impact on human lives.

Mathematics is generally perceived as a rather difficult domain to learn and master. Many analysts/designers do not want complex mathematics in order to capture a system's behavior because they think they might run into difficulty in achieving their goal easily.

Object Oriented System Design (OOSD)

Development of systems is progresses based on objects and actions which associates objects according to OOSD. Coad and Yourdon (1990, as cited in Dastbaz, 2002) explained an object as:
An abstraction of something in a problem domain, reflecting the capabilities of the system to keep information about it and interacts with it or both.

There are *three different object oriented* system design. They are;

- Object Modeling Technique: This technique has three phases that are analysis, design, and implementation. In analysis phase, the domain of problem is determined. The second phase, the result of analysis is structured and at the last phase the program utilized in target situation.
- OOSE Objectory: This method has three phases: Analysis, construction, and testing phase. First, the requirements of the system are determined. Later, information gathering from requirements phases are used for analysis or definition of domains of objects. The model of the objects and associations are produced in construction phase.
- Booch Method: It is also known as Object Oriented Design and Attributes. It provides step – by – step directions about design of a system. The steps start with identification of classes and objects then their semantics and relationships and finally implementation of them. While classes and objects are developing, they are packaged or transformed to modules diagrams. It can be also seen as incremental and iterative procedures.

Object Modeling Technique (OMT)

This method relies on a 'bottom up' approach and views the design process as being divided up into three main phases including:

- Analysis phase: which models the problem domain;
- Design phase: where the result of the analysis phase are structured and;
- Implementation phase: which takes into account target language constructs.

OMT has three clear stages at analysis phase: object modeling, dynamic modeling and functional modeling. Object Modeling involves identifying classes and their relationships and building class diagrams, which show relationships with one another. Dynamic behavior of a class is described in Dynamic Modeling using state diagrams and event flow diagrams. Functional Modeling involves identifying operations and putting them together by means of data flow diagrams. On the negative side it is weaker in its guidance during the design and implementation phases where it provides general guidelines and some heuristics.

Unified Modeling Language (UML)

Unified Modeling Language (UML) is a modeling language for object-oriented system development, which unifies concepts of many object-modeling languages. UML, since its adaptation by Object Management Group (OMG), the leading organization for standards within the object-oriented field, in 1997, has become a standard for modeling language. UML consists of three key elements:

- A formal meta model;
- A graphical notation;
- A set of idioms of usage;

In UML, primarily, there are nine different diagrams to capture the static, dynamic and architectural behavior of the system model. Case diagrams are used to capture requirement of a system. Class and Object diagrams are used to capture static structure of the system. Activity, Sequence, Collaboration and State diagrams are used to capture the dynamic behavior

Although UML is not meant for modeling structured software development, it has an ability to replace some aspects of structured system modeling. For example, UML use case diagrams can explain what their counterpart data flow diagrams do but only better. It seems so natural to think top level view in use cases where 'actors' can be seen as proper role players rather than 'source' or 'sink' in structured system. Similarly 'state transition diagrams' in UML can capture dynamic behavior of a system more effectively than their structured modeling counterpart 'entity life history' because the former captures every state of every class.

UML can be used in the different phases in the system development life cycle, from the requirement specification to the testing of an end product. It can be used to describe systems without any software too. UML can be used to model information systems, technical systems, embedded real-time systems, distributed systems, system software, business systems and the list is endless. Forward and reverse engineering is possible with UML. This clearly shows how versatile UML is.

OOSE Objectory

Objectory method has three phases: Analysis, construction and testing phase. The Requirement phase uses a natural language description of what the system should do to build three models (use case model, domain model and user interface descriptions). The Analysis model is a refinement of the domain object model produced in the requirements phase. The construction phase refines the models produced in the Analysis phase.

Booch Method

Booch Method is also known as Object Oriented Design with Attributes (OODA) and provides a step-by-step guide to the design of a system. These steps first identify classes and objects then their semantics and relationship and eventually implement them. During these steps class diagrams and object diagrams are

produced and packaged together by module diagrams. OODA views the design process as a highly incremental and iterative process.

Fusion

Developed by Hewlett-Packard, Fusion is based on experiences of several initial methods and uses a large number of model diagrams. This method claims to be a 'fusion' of the good concepts of other methods. It has improved ideas and techniques for the specification of operation and interaction between objects. Fusion has three phases: analysis, design and implementation. Each phase has a set of detailed steps and the output of one step acts as the input for the next.

Object Oriented Analysis and Design (OOAD)

The OOAD is based on a theoretical foundation, consisting of logic and set theory and attempts to integrate the static and dynamic aspects of OO analysis. In this method object flow diagrams are used to model high-level processes, event schemas are used to describe object behavior, and object schemas are used for describing static object types and their relationships.

Components of E-Learning Applications

Primary Components Of E-Learning Applications

A successful e-learning environments is the product of good decision made while design process. There are indispensable components of e-learning. In design process, designers attempts to combine these components. Good relationship and support among the components is the first step of effective e-learning applications.

The primary components of e-learning applications can be listed as;

- Learner motivation,
- Learner interface,
- Content Structure,
- Navigation,
- Interactivity.

Learner Motivation

Once e-learning is emerging, there are too many studies conducted to show its impact on learner outcomes. Motivation and e-learning interaction is one of the most repeated study types. Therefore, based on these observations or studies, we can conclude that motivation;

- Is a type of energy source for human being for learning activities to be completed,
- Is trigger impact on learners' energy for learning into two direction either positive or negative,
- Is a filtering tool to elect irrelevant stimuli which can be reason for inhibiting learning,
- Is a supporter to increase retention of new information,
- Is an encouragement tool in order to synthesis new information,
- Can create a relationship net to be considered and evaluated,

- Can be a bridge between old and new knowledge.

Learner Interface

Interface is a tool that supports communication between computer and user. It makes computers meaningful and usable to ordinary computer users. Without any interface design, e-learning courses are seemed worthless to learners.

Software interface and their usability that defined as learner satisfaction from an interface with respect to some dimensions, such as easy to use, memorability, relevancy, and etc. Usability and effectiveness of interface is just a huge field where there are lots of literatures. A good interface should support users with zero error. In other words, users cannot be suffered from interface while they are attempting to do a task, such as web based training course. In some situations, users are easily adapting themselves to interface although it is hard to use. It is named as learned helplessness. However, interface of an e-learning course should just provide the course content to users. Interface of a course is another concern for researchers among e-learning studies.

At first sight, people cannot care how computer works in their internal system. However, there should be something to make internal working principles meaningful to ordinary people. Interface is appeared here.

Novice or intermediate computer users generally have some worries about computer while they are using them. At this point, good interface plays an important role to prevent these worries. If a person wants to take a course from internet and have some worries about computer knowledge, designer should prepare interface whose usability seems to be convincible to these people.

The Interface Is the Computer

What people see of a computer is its external skins or interfaces-the keyboard, the mouse, and the options presented to them on the screen. Learners couldn't care much less how the computer works internally or how much effort it took to build instructional applications. Rather, they care about what they can do through reasonable effort with their computers and how interesting learning exercises are.

The lack of greater technical knowledge worries many computer users, however, and they feel a continuing risk of doing something stupid-or worse, something damaging. Interfaces provided for controlling the machine often contribute to this sense of insecurity. With many interface designs, for example, users wonder if there are unknown options that could make their work much easier or simple ways to fix problems they are having. They suspect there are features they want, but they can't find them and don't know what they are called. So they muddle along somewhat anxiously within the realm of known procedures. Help systems frequently infuriate people as much as they actually help, and users resign themselves to working within a set of familiar features.

To be more productive and comfortable, people need interfaces that relate to how and why they use a computer. They need interfaces that make options clear and understandable, rather than hidden, unintelligible, and exasperating. Product designers continuously strive to meet exactly these goals in addition to providing efficiency

and ease of learning. It is more difficult than it looks, but nevertheless an extremely important endeavor. A single interface weakness, just one, can lead to widespread user anxiety and discomfort.

Why should a single weakness have a far-reaching effect? Users look for patterns or conventions among controls and options provided. Conventions reduce the number of unique protocols that have to be remembered and usually provide helpful expectations of how unused options also work. The assumption users make is that the options work similarly to other related options or in exactly the same way as identically named options in other places.

When the consistency of conventions is broken in even a single instance, learners become uncertain about whether other conventions are also inconsistent. Every convention becomes immediately suspect. The software seems harder to use, confidence decreases, and many users barricade themselves within a subset of options that have proven reliable and at least minimally sufficient.

The Primary Responsibilities of Learner-Interface Design (Usability)

If learner interface is so important, it should have some responsibilities. It does support not only navigation, interactivity, communication, and information retrieval but also success of all components of the e-learning application. They are can be listed as;

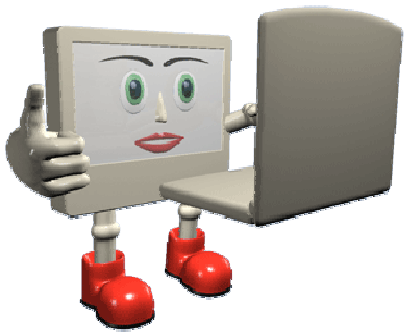
- A good interface should *minimize burden of memory*. In other words, user should not memorize each step while using it. Users should use minimum energy to use interface.
- A good interface should be *error free*. Each error is the hindrance to your e-learning activities. They can also influence in motivation of learners.
- A good interface should *minimize efforts.* Efforts using computer are minimized as much as possible. The magnitude of effort should be arranged based on learner levels. Hence, some of users are open to challenge and discover so more challenging interfaces encourages these kinds of learners to spend more efforts. However, novice users who cannot feel free to use computer system easily are not open to challenge. Therefore, more structured and effortless interfaces are more suitable to these kinds of people.
- A good interface should have *promoting features*. To minimize memory burden, some of tools can be hidden in an interface. However, some procedures cannot be memorized easily so they should be easy to access by users. An interface should support these kinds of things, such as hints or tips for some features.
- A good interface should *contribute to the learning experiences*. It should support as much as possible learning activities within the course. Interface can combined with context in some situations to enhance learning experiences.
- A good interface should give *feeling of control to users*. They really are aware of what they are doing.
- A good interface should *encourage a bit amount anexity and discomfort to provide positive stress on users.* It also helps to persist motivation level.
- A good interface should *support challenges for the purpose of learning.* Challenges do not embedded into interfaces' directions.

The most important reason why designers have to consider interface design is influence of it on learning outcomes. Designers should develop error free computer based environments that is web sites or pages in e-learning. In other words, the fundamental duty of a participants in e-learning applications is just learning. Your applications should provide just learning activities. The other components do supplement learning process.

Importance of Good Interface Design for e-Learning

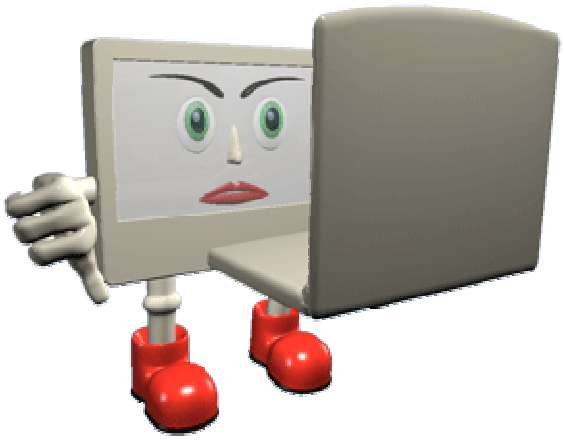
If good user interface design is important anywhere, it's critically important in e-learning. Why? It is a daunting challenge to get people to change their behavior, which may include changing habits, perceptions, and values, as well as

acquiring knowledge and developing new skills. It's less of a challenge if you have instructional design education, training, experience, and talent, but even for those select pros, it's always a difficult challenge. To succeed, it's important to gain a high percentage of the user's attention.



Effects of Poor Interface Design

Poor user interface can;



- Repeatedly and frequently distract the user's attention
- Make text difficult to read and graphics ineffective
- Cause branching to the wrong information or exercises
- Confuse learners about their progress and their location within the application
- Make useful activities too bothersome to complete
- Obscure access to needed information
- Make comparisons difficult
- Slow interactions
- Debilitate feedback (Allen, 2003).

Poor interface design can impact learners' performance while they are attending e-learning applications. Hence, they spend their efforts to solve how interface works. It takes time and means that learners spend more time on learning activities.

Content Structure and Sequencing

Content Structure and Sequencing

Determination of content structure and sequencing is the most important concern of instructional design process of e-learning applications. In an ID process, designers attempt to answer what content to include, how to use it, and in what order. In other words, a designer first select which content is appropriate to students, second transform it into the most suitable format, and finally put it in a continuous structure.

What is content?

Content is one of the discussed topic of instructional design. There is not exact consensus about content because content of a course can vary from one situations to other. One cannot express some strict standards. The best indicator of content about a course is degree of reflection of students' needs.

Allen (2003) asserted different types of content definitions. These are presented in the following table:

Information-based	Content is all the information, such as facts, concepts, and procedures to be learned. A detailed outline, for example, would summarize the components.
Objectives-based	Content is a collection of learning objectives specifying behavioral outcomes. For example, "At the conclusion of this learning activity, students will be able to name four of the planets in our solar system
Media-based	Content is all the text, graphics, videos and other multimedia components of an instructional

	application.
Experience-based	Content is the sum of all instructional components in a learning application, including the learning objectives, media, interactions, and assessment activities.

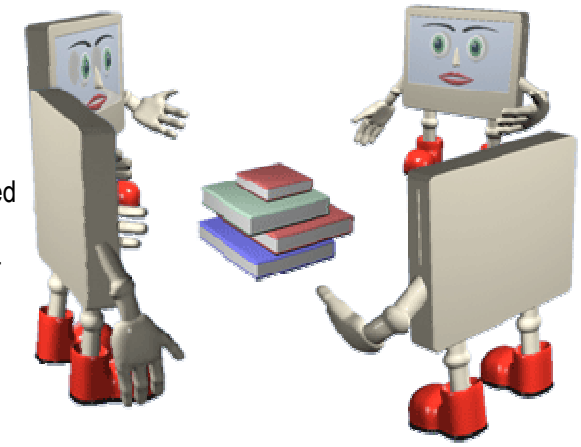
Who Cares?

Lack of a standard definition may not be a critical issue in itself, but it is clear there are many misunderstandings about e-learning-what's good, bad, possible, impractical, and so on. Varying definitions of content may cause some of the misunderstandings, since assumptions of what a person means by content may be quite incorrect.

It depends on which definition of content is used. Those with an information-based definition would be thinking that the manager wants little spent on interactivity so that careful presentation of information can be accomplished, perhaps with good navigation controls.

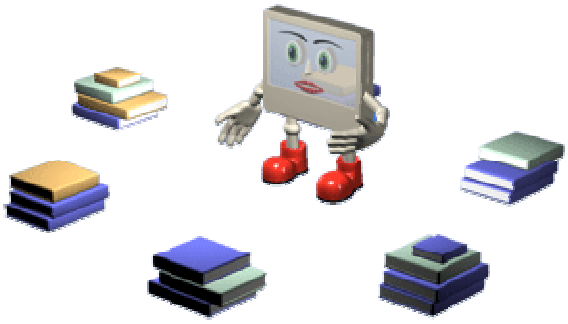
Content-Centric Design

The content should be presented to learners as clearly as possible. Content should be taught meaningful and memorable experiences. Main disadvantage of content-centric designs is giving less attention on learners' experiences. Content-centric design is structured based on subject-matter experts' point of view, rather than learners' point of view. Sometimes, learners' point of view is considered to illuminate or declaration of some points rather than determining sequencing of content. Purely designed content-centric design applications may not work well for adult education environments. In addition to this, if designers cannot provide memorable and meaningful activities, application becomes a problem rather than a solution to learners' needs. Therefore, while using content-centric designs, designers should determine each step carefully.



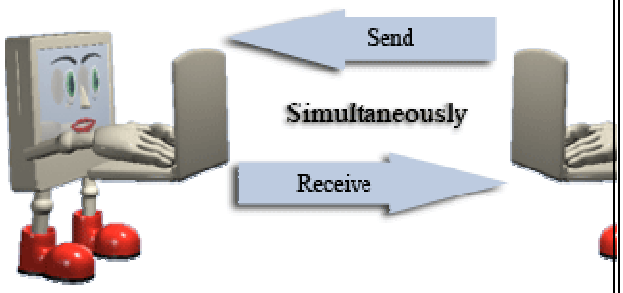
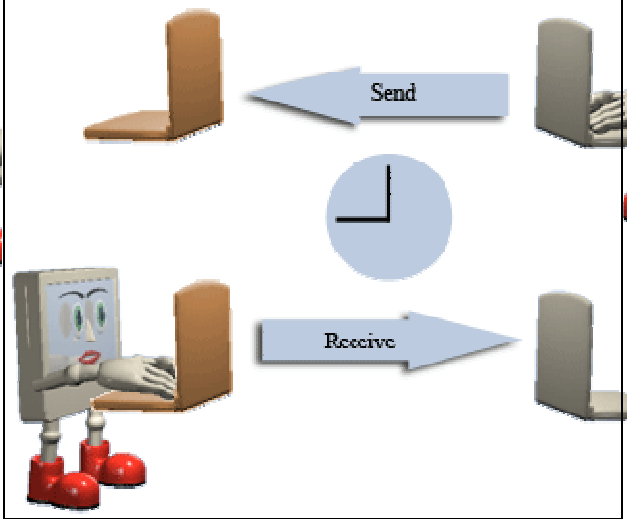
Learner-Centric

Presenting content completely without any errors cannot end with successful results. Complete presentation does not make any sense if it cannot fulfill learners' needs. Learner-centric designs focus on creating events and activities based on learners. Learners have some responsibilities in e-learning application. Learners should figure out something by themselves. If they can do that, they are awarded; otherwise they are supported by aids. After completion each task, learners should put new information into their current knowledge. It is possible with providing learners with related or subsequent tasks. Therefore, learners can find some hints or connection point among information.



Synchronous or asynchronous?

One of the important decisions waiting for designers is to determine presentation of e-learning application either synchronous or asynchronous. Another possibility is combination of them. The main question is "Can the learner control his/her learning time?" Answer of these questions can help designers to give a decision. The application term synchronous and asynchronous is more suitable for individual learning environments than groups.

Synchronous: Everyone must involve an activity at the same time. Such events recall as real time or live events. They could be chat sessions, videoconferencing, screen sharing, and whiteboard.	*Asynchronous:* These kinds of activities can allow learner to do activities at any time. For example, posted web messages, forums, e-mails are asynchronous tools.
	

Sequencing for Learning

Content-centric and learner centric designs use very different content sequencing. For example, in content-centric designs, subject matters use three sequencing techniques that are simple to complex, chronological, or hierarchical. These sequences are stem from theoretical analysis of content.

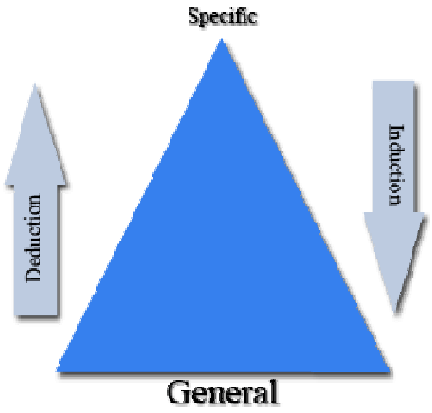
In learner-centric designs, the key issue is learner so sequencing should be made based on learner needs. Learners expect relation points between topics. They are also eager to learn in a surprised ways. Designers can use some sequences that are known to unknown, misconception to latest technique, and goal decomposition. The main purpose of these sequences is supporting learning to reach vital performance goal. It can be done by demarcating main goal to the sub-tasks.

In learner centered sequences;

- Learners initial competencies are important. They should be determined at the beginning of the e-learning application.
- Content should be divided into meaningful chunks that are based on performance events.
- Content should advance in a meaningful shape. However, learner cannot easily understand rationale of the sequencing. The dose and place of challenge is important here.
- Content should provide as many as possible tasks whenever it is wanted if learners demand them. They could be more difficult or more based on interactive events.
- Learners should be allowed to review their progress.

Structuring Events

We can use general two types of structuring for e-learning activities. They are inductive and deductive structuring. In inductive structuring, learner can reach the overall picture of instructed topic from the small and related pieces. This type of structuring is more suitable for learner centric designs. On the other hand, content can be structures deductively. In deductive structuring, learners should understand concept from macro level to micro level. Learner should first main understanding of concepts and then they should investigate main concepts in detail.



Navigation & Interactivity

Learner Needs Addressed by Navigation

Navigation can support the serving of an e-leaning application. The more strength navigation structure could affect effectiveness of e-learning application in a positive ways.

Allen (2003) stated that navigation in e-learning applications can provide many valuable services, including:

- The ability to preview and personally assess:
- What can be learned
- How valuable it will be
- How much time it will take
- How difficult it will be
- Once into the application, the ability to determine:
- How much you have accomplished
- How much remains to be learned
- Overall, the ability to:
- Back up and review
- Back up and try different answers or options (p. 92).

Reactions of learner to e-learning application are critical effect on results. The first impression is so important that it can produce wonderful or terrible products. Navigation is one component of first impression. Moreover, persons' expectations from the course could be met by designing effective and useful navigations. Otherwise, not fulfilled expectations can hinder motivation level of learners. Navigation is a tool to provide election system. Therefore, people can make individual decisions about their learning. Good navigation does not mean that take a person from point A to point B. Navigation should provide both the path of e-learning application and learning experiences.

Electronic performance support systems (EPSSs)

Electronic performance support systems (EPSSs) are a good example of applications that focus heavily on navigation. EPSSs provide real-time information, prompting, and data processing tools to guide performers. In contrast to e-learning applications which help users perform based on internalized information and practiced

skills, EPSS logic provides the expertise needed to guide users as they work. Users may remain dependent on the system for support, but such dependence may not be detrimental or even undesired in many situations, such as those where data entry is part of the transaction anyway and where procedures change frequently or where oversights and errors might have grievous consequences.

Navigation Can Help or Hinder Learning

It's really frustrating to be uninformed and not know what to expect. Unless initial impressions are extremely positive, perhaps elevated by a fascinating opening or by reputation or trusted endorsement, people naturally become suspicious that lack of information means bad news. Expectations of a delightfully beneficial experience are replaced by doubt, if not dread. Not a good start.

Expectations set initial attitudes. Attitudes, in turn, assist or hinder the effectiveness of e-learning. With all the challenges facing us in the attempt to get people to do what we want them to do, a positive attitude is definitely something to foster. In many ways, all our efforts to shape behavior are much like processes of sales. We need to get people to want what we have to sell before we can expect them to buy. Constant focus and restatement of benefits is important. Navigation provides one of the ways people can pick up and examine our products and distinguish their benefits. It's the packaging that can cause our product to go home with the learner or be put back on the shelf for later consideration.

As you can see, navigation in e-learning is far more than just getting from point A to point B. Navigation facilities provide a major component of the learning experience. Good capabilities not only enhance the power of presentation and interactive components, but also provide learning experiences directly by allowing learners to compare and research.

Instructional Interactivity

The purpose of instructional interactivity is to support people's learning if it does not work for a long time. In other words, it attempts to warn you for learning process. It elicits background information, knowledge, and skills. It seems a kind of sports training. Before starting new session, the team has to conduct training.
Allen (2003) mentioned instructional interactivity is not the same as:

- Navigation
- Presentation
- Buttons
- Scrolling
- Browsing
- Information retrieval
- Animation
- Morphing
- Video

Instructional interactivity is focusing internal events, such as recall, classification, analysis, and decision making rather than observable events. For instance, clicking a mouse or dragging an icon.

Beneficial Activities

In e-learning application, learner should be supported by some activities that help them to be active. Someone cannot teach new knowledge to those persons. They have to learn new knowledge by themselves. Main role of instructional design is to determine activities that result in learning and designing structures that attracts learner to them.

Horton (2001) proposed some sample instructional interactivity that is useful for designers.

- Webcasts
- Presentation sequences
- Drill & Practice activities
- Scanverger Hunt
- Guided Research
- Guided Analysis
- Team Design
- Brainstorming
- Case Studies
- Role-playing scenarios
- Group Critiques
- Virtual Laboratories
- Hands-on Activities
- Learning Games

Characteristics of Interactivity

- Interaction can force learner to commit an answer or to perform a task before receiving a feedback.
- Learners have a chance to apply their skills in interactive environment.
- Behind the electronic curtain, system can direct learners based on their performance.
- Learners have an opportunity to make mistakes. Learning by making mistakes is one of the best ways of learning in safe situations.
- Interactive Systems can provide large scope of sources at the same time.
- Interactivity can be also support multimedia, automated tracking.
- Interactivity can be also used as a communication tool for learning or teaching purposes

Samples of Instructional Interactivity (1)

Activity	Description	When to use it
Webcasts	Many distributed learners participate fully in a conventional training event transmitted by a network.	To teach material best taught by traditional classroom activities, especially ones that require extensive interaction between the instructor and learners.
Presentation sequences	Learners read, listen to, and watch carefully crafted explanations in a Web browser.	To provide a consistent high-quality explanation to all learners.
Drill & Practice activities	Learners repeatedly practice applying specific knowledge or a well-defined skill.	To help learners memorize facts that they must be able to recall without hesitation.
Scanverger Hunt	Learners find reliable sources of information on the Internet or their corporate intranet.	To make learners more self-reliant by having them locate reliable sources of information on the subject they are studying.
Guided Research	Learners gather, analyze, and report on information.	To teach learners to conduct informal research on a subject. This activity is especially valuable for learners who will have to conduct informal research as part of their job.
Guided Analysis	Learners analyze data to evaluate its validity, spot trends, and infer principles.	To teach a formal analysis technique or to guide learners to discover trends and principles for themselves.
Team Design	Learners work as coordinated teams to produce a single design or to solve a complex problem.	To teach design skills that are applied as part of a team or to teach basic teamwork skills.
Brainstorming	Distributed learners work together to generate creative solutions to a problem or to accomplish some other goal.	To teach brainstorming in its own right or as part of a course involving problem solving, creative thinking, or team design.
Case Studies	Learners study a meaningful, detailed example of a real-world event, process, or system to abstract useful concepts and principles.	To teach complex knowledge that cannot be reduced to a simple formula. To use specific, concrete particulars to teach abstract, general principles.
Role-playing	Learners adopt assigned rotes in	To teach subtle interpersonal skills and to reveal the

scenarios	simulations involving complex interpersonal interaction.	complexity of many human endeavors.
Group Critiques	Learners receive and react to the criticisms from their peers. Learners submit a work that others in the class critique.	To teach learners how to use critical comments of others to improve their own work and how to offer helpful criticism of the work of others.
Virtual Laboratories	Learners conduct experiments with simulated laboratory equipment.	To prepare learners to operate real laboratory equipment or to guide them to discover principles and trends on their own.
Hands-on Activities	Learners perform a real task outside the lesson.	To teach hands-on tasks and to show learners how to apply abstract knowledge gained in other activities.
Learning Games	People learn by playing.	Learning games are computer simulations that let learners practice a highly interactive task. To give learners experience performing a task without the risk or cost of the real activity.